Brian C. Rocca, S.B #221576
brian.rocca@morganlewis.com
Sujal J. Shah, S.B #215230
sujal.shah@morganlewis.com
Michelle Park Chiu, S.B #248421
michelle.chiu@morganlewis.com
Minna Lo Naranjo, S.B #259005
minna.naranjo@morganlewis.com
Rishi P. Satia, S.B #301958
rishi.satia@morganlewis.com
MORGAN, LEWIS & BOCKIUS LLP
One Market, Spear Street Tower
San Francisco, CA 94105
Telephone: (415) 442-1000
Facsimile: (415) 422-1001

Counsel for Defendants

Glenn D. Pomerantz, Bar No. 112503
glenn.pomerantz@mto.com
Kuruvilla Olasa, Bar No. 281509
kuruvilla.olasa@mto.com
MUNGER, TOLLES & OLSON LLP
350 South Grand Avenue, Fiftieth Floor
Los Angeles, California 90071
Telephone: (213) 683-9100

Justin P. Raphael, Bar No. 292380
justin.raphael@mto.com
MUNGER, TOLLES & OLSON LLP
560 Mission Street, Twenty Seventh Fl.
San Francisco, California 94105
Telephone: (415) 512-4000

Jonathan I. Kravis, *pro hac vice*
jonathan.kravis@mto.com
MUNGER, TOLLES & OLSON LLP
601 Massachusetts Ave. NW, Ste 500E
Washington, D.C. 20001
Telephone: (202) 220-1100

UNITED STATES DISTRICT COURT

NORTHERN DISTRICT OF CALIFORNIA

SAN FRANCISCO DIVISION

IN RE GOOGLE PLAY STORE ANTITRUST LITIGATION THIS DOCUMENT RELATES TO: *Epic Games Inc. v. Google LLC et al., Case No. 3:20-cv-05671-JD*	Case No. 3:21-md-02981-JD **DECLARATION OF DAVID KLEIDERMACHER IN SUPPORT OF GOOGLE'S OBJECTIONS TO PROPOSED INJUNCTION** Judge: Hon. James Donato

DECLARATION OF DAVID KLEIDERMACHER

1. I, David Kleidermacher, am Vice President of Engineering for Security and Privacy for Android and Made-by-Google Products and Services. Since I joined Google in 2017, I have been responsible for Android security and privacy engineering. I am familiar with Google's efforts to combat the harmful effects of malware on Android. I make this declaration based on personal knowledge. If called as a witness, I could testify competently to the facts stated herein.

Restrictions on Sideloading Warnings or Consent Screens

2. I understand that Epic has proposed that the Court should prohibit Google from warning users about the security risks of sideloading an app (*i.e.*, direct downloading from the internet or from an app store downloaded from the internet) unless Google has determined that an app is "known" malware or the app's developer has declined to participate in a generally available "notarization-like" process. Google does not have any generally available "notarization-like" process for Android apps so, in effect, Google would be limited to warning users about known malware unless it undertakes the burden and expense of creating a new "notarization-like process."

3. Limiting Google to warning users about "known" malware is a very significant limitation that will degrade user security. When a user attempts to download and install an app, it can be difficult or impossible to determine, at that moment in time, whether that app is malware. In some cases, an app can be identified as "known malware" if Google has previously reviewed that exact app and has classified it as malware. But bad actors constantly develop new variants of malware, some forms of malware are capable of altering themselves (a property called polymorphism), and some forms of malware are capable of obfuscating their code to defeat scanning. As a result, in many cases, it will not be clear whether a particular app is malware at the time of downloading and installation.

4. Today, if Google Play Protect—security software on Android devices that include the Google Mobile Services suite—does not recognize a sideloaded app as known malware, it may nonetheless use other signals about the app to inform the user that the app carries elevated security risk. For example, the app may use out-of-date software interfaces that do not incorporate the

latest Android security features or very few users may have installed the app. In other cases, Google Play Protect may not detect any signals that suggest that a particular app may be malware. Epic's proposal means that, in all these scenarios, Google would not be able to warn users about the risks of sideloading because it will not know that the app is actually malware. This is a poor outcome from a security perspective.

5. Government agencies and industry participants have also warned users about the risks of sideloading.

6. For example, Europol has warned users to "[b]e cautious of links you receive in email and text messages that might trick you into installing apps from third party or unknown sources." A true and correct copy of a Europol document titled "Mobile Malware: Tips & Advice to Protect Yourself," available at https://www.europol.europa.eu/sites/default/files/documents/Infosheet%20-%20Private%20Users.pdf (downloaded April 29, 2024), is attached hereto as **Exhibit A**.

7. Samsung has similarly warned users that "sideloaded apps from outside sources can be a little like the Wild West—unregulated and potentially hazardous. The reason? They may carry hidden malware designed to compromise your device or even your personal information." A true and correct copy of a Samsung webpage titled "What Is Sideloading (and Why Should You Care)?," available at https://www.samsung.com/uk/explore/life-hacks/what-is-sideloading-and-why-should-you-care/ (downloaded April 29, 2024), is attached hereto as **Exhibit B**.

8. Given the heightened risk of installing malware through sideloading, a user should be made aware of the risks of sideloading at the time the user decides whether to enable sideloading. In addition, Google should be permitted to warn users when a particular sideloading installation raises additional concerns (e.g. out-of-date software interfaces). Without these warnings, users could more easily install malware that may harm their device, violate their privacy, or steal their financial data or money, among other common harms.

9. I also understand that under Epic's proposal, Google will be permitted to display only certain narrowly-defined warnings or consent screens in connection with sideloading. In my experience, Android malware often relies on deceiving users into granting the malware the

permission to be installed or to take certain harmful steps. This deception is often called "social engineering." Limiting Google to presenting users with only narrowly defined app installation screens would make it more difficult to respond to the many ways in which malware attempts to deceive and harm users. These deceptive methods also often change as malware attackers evolve their strategy, making it particularly important to respond quickly to new variants of malware. For example, today, if Google learns that new variants of malware are abusing a certain Android functionality, it can take steps to warn users about that risk or require user additional consent before allowing a sideloaded app to access that functionality. Under Epic's proposal, I understand that Google will not be able to take such steps to protect users.

Restrictions on Innovative Security Features

10. Epic's proposal also has negative implications for other innovative security features that protect users. For example, Google has launched the Advanced Protection Program, which is a set of optional security measures intended to protect against targeted malware attacks—that is, malware attacks that are precisely aimed at specific users or groups of users. While any Google account holder may enable Advanced Protection, it is designed to address the unique and heightened risks faced by political dissidents, journalists, high-ranking government officials, and high-profile business executives. If a user has enabled Advanced Protection, all sideloading is blocked and the user is warned before installing any app from the Play store outside of a small set of pre-approved apps. Epic's proposals to restrict Google's Android security mechanisms to blocking "known malware" or apps that decline to participate in a security scanning process could effectively require Google to shut down the Advanced Protection Program's Android security features for users who have already opted-in. This would put individuals who face targeted attacks, like journalists and political dissidents, at a much higher risk.

11. Android OEMs also have the ability to innovate on security issues and safeguard their users from sideloading risks. To the extent that Epic's proposed injunction applies to OEMs, it would prevent OEMs from providing users with additional security features and protections. For example, Xiaomi, an Android OEM, has added an additional sideloading warning and imposes a mandatory ten-second countdown before the user may enable sideloading.

12. A true and correct copy of a screenshot of a sideloading warning on a Xiaomi Mi 10T Lite 5G Android device is attached hereto as **Exhibit C**.

13. Samsung has its own suite of Android security features, Samsung Knox, that works alongside the default features built into Android and Google Play Protect. Samsung has also recently announced an "auto blocker" feature that allows users to opt-in to block sideloading by default.

14. A true and correct copy of the Samsung webpage titled "Protect Your Device Your Way With Samsung Auto Blocker," available at https://news.samsung.com/global/protect-your-device-your-way-with-samsung-auto-blocker (downloaded April 29, 2024), is attached hereto as **Exhibit D**.

Requests from Government Agencies and Trade Groups

15. Despite Google's efforts to combat Android malware, significant security risks remain. Government agencies around the world have requested Google's assistance in addressing this threat by making it more difficult for malware to be installed on and to spread on Android devices. For example, Google has recently partnered with the Cyber Security Agency of Singapore (CSA) to release new technology for Android devices in Singapore that will analyze and automatically block the sideloading of apps that use sensitive permissions that are frequently abused for financial fraud.

16. A true and correct copy of the Google webpage titled "Piloting new ways of protecting Android users from financial fraud, " available at https://security.googleblog.com/2024/02/piloting-new-ways-to-protect-Android-users-from%20financial-fraud.html (downloaded April 29, 2024), is attached hereto as **Exhibit E**.

17. Google developed this technology in response to a request from the CSA to help stem a rising tide of malware harming Singaporean users. Epic's proposal would prevent Google from blocking this high-risk sideloading in Singapore and elsewhere.

18. A true and correct copy of an August 22, 2023 email from Henry Tan, Deputy Director, Cybersecurity Engineering Center, Cyber Security Agency of Singapore is attached hereto as **Exhibit F**.

19. Other government agencies have similarly requested that Google assist in responding to malware threats. For example, the Thai government's Ministry of Digital Economy and Society has requested that Google's Singapore-focused technology also be deployed in Thailand.

20. A true and correct copy of a February 15, 2024 letter from Prasert Jantararuangtong, Minister of Digital Economy and Society of Thailand, is attached hereto as **Exhibit G**.

21. The Brazilian Federation of Banks has similarly requested that Google "restrict sideloading entirely or make it significantly more difficult" because of "a worrying trend of financial scams utilizing malicious mobile apps" where scammers "trick users into sideloading apps containing malware."

22. A true and correct copy of a December 19, 2023 email from Walter Tadeu Pinto de Faria, Deputy Director of Services at the Brazilian Federation of Banks (FEBRABAN), is attached hereto as **Exhibit H**.

<u>Notarization</u>

23. I understand that Epic proposes that Google could also block or warn users regarding apps that refuse to go through a "notarization-like" review process. Epic's proposal does not contain any details regarding this process. In any event, building an app review process for apps distributed outside of Play would require a significant investment of engineering and operational resources. While it is difficult to estimate the precise cost of building or maintaining such a system without detailed specifications for what the system would entail, a basic notarization system would cost at least tens of millions of dollars per year to operate, and likely over $100 million per year, and even at that level of spending the system would not provide the same degree of safety as the app review process employed by the Play store today.

24. Building such a system would also take a significant amount of time and could not realistically be implemented in a few months. For example, Google would need to invest in developing and maintaining new technology and infrastructure to allow non-Play developers to submit apps to Google and to allow Google to accurately assess the risk of those apps. And to the

extent that this review process requires human review to provide adequate security assurances, Google would need to employ additional security analysts to review apps distributed outside the Play store.

25. A new notarization system would also carry substantial risks for Google, users, and developers because it is not clear that such a system could ever achieve the same level of safety as the Play store. I am not aware of any other consumer operating system that has developed a notarization system that would provide the same level of safety as the Play store.

Pre-Installation

26. I also understand that Epic has proposed prohibiting Google from asking OEMs not to pre-install any app or app store, regardless of the security, privacy, or user experience risks that a particular app or app store could create. For example, Google has policies that require Android OEMs who preload Google Mobile Services (GMS) apps to ensure that all preloaded apps on the OEM's devices safeguard user privacy, are not malware, do not contain illegal or harmful content, disclose information regarding who created the app, and seek user permission for certain sensitive operations. Preventing Google from enforcing these and other similar policies would harm users. Some OEMs, in the interest of cutting costs or raising revenue from placement fees, may elect to include apps that they have not rigorously vetted or to include apps that compromise user security or privacy.

27. In fact, consumer privacy organizations have advocated for Google to take steps to reduce the prevalence of harmful pre-installed apps on Android devices. In January 2020, Privacy International issued an open letter to Google's CEO requesting that Google take more aggressive actions to address rising malware that is preinstalled on Android devices by OEMs.

28. A true and correct copy of the Privacy International webpage titled "An open letter to Google," available at https://privacyinternational.org/advocacy/3320/open-letter-google, (downloaded April 29, 2024) is attached hereto as **Exhibit I**.

Security-Sensitive Android APIs

29. Android includes certain APIs that provide powerful and security-sensitive functionality that can pose a significant risk of harm to a user or her device. For example, there are

APIs that allow an app to perform a factory reset on a device (i.e., delete all of the apps and content on the device), delete other apps, place phone calls, and connect to nearby Bluetooth devices. In the wrong hands, these APIs could be used to harm users, such as by deleting the phone dialing app when a user tries to call 911, placing calls to premium numbers that incur a charge on the user's phone bill, or surreptitiously connecting a user's device to an illicit Bluetooth device to track the user's physical location.

30. OEMs who make Android devices often need access to these sensitive APIs in order to configure their devices and provide them with custom functionality. For example, an OEM may design the device to automatically connect over Bluetooth to certain compatible hardware (such as headphones) or an OEM may include software that manages factory resets that, for example, allows users to remote reset a lost device. That functionality would not be possible if a user were required to grant permission (because the device is lost) but would be extremely harmful if it could be accessed by any app without the device OEM's approval.

31. Because these APIs are highly sensitive, Android includes two classes of these APIs: one class that is available to any developer if the user grants the developer permission to use it and a second class that is available only to the OEM who made the Android device. This latter OEM-only class does not require explicit user permission in order for the OEM to access the API. This framework balances competing interests by allowing OEMs to build advanced functionality for their devices but also protecting users from bad actors who would abuse these APIs.

32. If Google were required to provide all apps—not just the OEM's preinstalled apps—with access to all APIs on the device, Google would either have to remove the OEM-specific APIs (which would make it more difficult for OEM's to design advanced functionality) or provide all apps with access to the OEM-specific APIs (which would allow malicious apps to harm users).

The Android OS Development Process

33. Epic is requesting certain remedies that will necessarily require changes to the Android OS. For example, Epic is requesting that Google shall be required to provide users with the ability, subject to a one-time user permission, to change the ownership for any or all of the

apps installed on their device, such that a third party app store becomes the update owner for those apps. And, as discussed above, Epic is requesting numerous changes to the way Android approaches security including (1) changes to Android's security warnings; (2) modifications to the Android operating system to support a "notarization-like" process; and (3) modifications to Android's permissions system for highly sensitive APIs.

34. These proposals would require material changes to Android OS, as they go beyond what the OS is currently designed and coded to do.

35. Developing and releasing the source code for a new version of the Android platform is a complex process that takes a year or more. The Android Open Source Project (AOSP) maintains a complete software stack that can be adapted by OEMs to run on their own hardware.

36. During the OS development process, Google maintains the current stable version of Android (e.g., Android 14 that was released last year) separate from any unstable experimental work on potential future releases. At any given moment, there is a current latest release of the Android platform that device builders and contributors work on to improve and prepare for mass distribution (e.g., fixing bugs, launching new devices, experimenting with new features, etc.).

37. In parallel, Google works internally on subsequent versions of the Android software. When the next version is ready, the software is published to the public source tree and becomes the new latest release. First, the new software is customized by OEMs, and put into a system image for a device. Second, devices are put through various forms of certification, including, among other things, government regulatory certification for each of the regions the phones will be deployed. The code also goes through mobile network operator testing. This is an important and time-consuming phase of the process, because it helps detect software bugs which could significantly undermine the integrity and viability of the software release.

38. Third, when the release is approved by regulators and mobile network operators, the manufacturer begins mass producing devices, and we begin releasing the source code.

39. Simultaneous to mass production, the Google team executes several involved processes to prepare the open-source release that meets the requisite standards for broad

consumption. These efforts include making final API changes, updating documentation (to reflect any modifications that were made during qualification testing, for example), preparing an SDK for the new version, and launching the platform compatibility information.

40. The entire process of developing, testing, and releasing a new version of Android typically takes a year or more.

41. Once a final version of Android has been publicly released, Google does not typically update old versions of Android other than to release critical security patches. This is important because changing the functionality of old versions of Android can cause apps to unexpectedly malfunction if developers did not build their apps to anticipate the new changes. Google also does not have any mechanism to force updates to old versions of Android because OEMs control the availability of updates on their devices.

42. With respect to app ownership (for the purpose of updating apps), the latest version of Android OS, named Android 14, was released in October 2023. The release of Android 14 followed the general outline of development steps described above. In that release, Google introduced several PackageInstaller APIs that allow app stores to improve their user experience. For example, Google enabled new functionality related to the way installers (including app stores) can update apps installed on a user's device. Specifically, Android 14 allows an installer of an app to indicate that it intends to be responsible for future updates for that app. Only the update owner is permitted to install automatic updates to the app. Update ownership enforcement helps to ensure that users receive updates only from the expected app store, i.e., the original installer.

43. Under Android 14, any other installer must receive explicit user approval in order to install an update. If a user decides to proceed with an update for an app from another source, and provides explicit approval, update ownership is lost by the original installer. At that point, any installer can provide updates to the installed app.

44. My understanding is that Epic is seeking to fundamentally change this app update ownership protocol, which would require significant and burdensome changes to Android OS and functionality that Android has never supported. The approach would be different from what Google released in Android 14, which requires an app-by-app consent process which, among other

things, preserves user choice and, assuming the user provides consent for a given app, does not permit a change in update ownership from Store A to Store B, but rather removes update ownership permission from Store A. Thus, Google, and its Android contributors and partners, would need to create new software, test it, and launch it under the Android protocol set forth above.

I declare under penalty of perjury that the foregoing is true and correct. Executed on this 1st day of May 2024 in Palo Alto, CA.

DocuSigned by:

A955A84AE7C7405...

David Kleidermacher